**Importance of below 4 demons in job execution**

**1)Name Node:**

* The NameNode is the centerpiece of an HDFS file system
* It keeps the directory tree of all files in the file system, and tracks where across the cluster the file data is kept
* Client applications talk to the NameNode whenever they wish to locate a file, or when they want to add/copy/move/delete a file
* The NameNode responds the successful requests by returning a list of relevant [DataNode](https://wiki.apache.org/hadoop/DataNode) servers where the data lives
* The NameNode is a [Single Point of Failure](https://wiki.apache.org/hadoop/Single%20Point%20of%20Failure) for the HDFS Cluster

**2) Data node :**

* A DataNode stores data in the [[HadoopFileSystem](https://wiki.apache.org/hadoop/HadoopFileSystem)]
* A functional filesystem has more than one DataNode, with data replicated across them.
* On startup, a DataNode connects to the [NameNode](https://wiki.apache.org/hadoop/NameNode); spinning until that service comes up
* It then responds to requests from the [NameNode](https://wiki.apache.org/hadoop/NameNode) for filesystem operations.
* Client applications can talk directly to a DataNode, once the [NameNode](https://wiki.apache.org/hadoop/NameNode) has provided the location of the data

**3) Resource Manager :**

* **ResourceManager (RM)** is the master that arbitrates all the available cluster resources
* IT helps to manage the distributed applications running on the YARN system
* It works together with the per-node **NodeManagers (NMs)** and the per-application **ApplicationMasters (AMs)**
* **NodeManagers** take instructions from the ResourceManager and manage resources available on a single node.
* **ApplicationMasters** are responsible for negotiating resources with the ResourceManager and for working with the NodeManagers to start the containers.

**4) Node manager :**

* The NodeManager (NM) is YARN’s per-node agent
* It takes care of the individual compute nodes in a Hadoop cluster
* This includes keeping up-to date with the ResourceManager (RM), overseeing containers’ life-cycle management
* Monitoring resource usage (memory, CPU) of individual containers, tracking node-health
* Log’s management and auxiliary services which may be exploited by different YARN applications.